

## BAB IV

### PENGUJIAN DAN EVALUASI SISTEM

Berikut ini akan dijelaskan tentang pengujian perangkat keras dan perangkat lunak dari alat yang dirancang, serta integrasi keduanya. Tujuan dari pengujian disini adalah untuk mengetahui apakah setiap perangkat sistem telah berfungsi dengan baik.

#### 4.1. Prosedur Pengujian Perangkat Keras

Pengujian terhadap perangkat keras meliputi pengujian pada PPI, DAC, ADC, *Driver Motor DC*, *Tachogenerator*, Rangkaian Penyearah dan Rangkaian Pelemah. Pengujian perangkat keras mengacu pada perancangan perangkat keras seperti yang telah dijelaskan pada Bab III.

##### 4.1.1 PPI

Pengujian PPI 8255 berkenaan dengan *input - output* ketiga *port* yang ada. Untuk mengujinya dibutuhkan program sederhana. Berikut adalah program untuk menguji kelayakan setiap *port* dalam Turbo C++ :

```
#include <stdio.h>
#include <dos.h>
#include <conio.h>
#include <iostream.h>

#define PortA 0x3e0
#define PortB 0x3e1
#define PortC 0x3e2
#define Cw 0x3e3

int main(void)
{
    outportb (Cw,0x80); // ke tiga port PPI diset untuk Output semua
    do
    {
        outportb (PortA,0x01); // lampu ke- 8 menyala
        delay(500);
        outportb (PortA,0x02); // lampu ke- 7 menyala
        delay(500);
        outportb (PortA,0x04); // lampu ke- 6 menyala
        delay(500);
        outportb (PortA,0x08); // lampu ke- 5 menyala
    }
```

```

delay(500);
outportb (PortA,0x10); // lampu ke- 4 menyala
delay(500);
outportb (PortA,0x20); // lampu ke- 3 menyala
delay(500);
outportb (PortA,0x40); // lampu ke- 2 menyala
delay(500);
outportb (PortA,0x80); // lampu ke- 1 menyala
delay(500);
outportb (PortA,0xFF); // semua lampu menyala
delay(500);
outportb (PortA,0x00); // semua lampu padam
/* Port B */
outportb (PortB,0x01); // lampu ke- 8 menyala
delay(500);
outportb (PortB,0x02); // lampu ke- 7 menyala
delay(500);
outportb (PortB,0x04); // lampu ke- 6 menyala
delay(500);
outportb (PortB,0x08); // lampu ke- 5 menyala
delay(500);
outportb (PortB,0x10); // lampu ke- 4 menyala
delay(500);
outportb (PortB,0x20); // lampu ke- 3 menyala
delay(500);
outportb (PortB,0x40); // lampu ke- 2 menyala
delay(500);
outportb (PortB,0x80); // lampu ke- 1 menyala
delay(500);
outportb (PortB,0xFF); // semua lampu menyala
delay(500);
outportb (PortB,0x00); // semua lampu padam
/*
Port C */
outportb (PortC,0x01); // lampu ke- 8 menyala
delay(500);
outportb (PortC,0x02); // lampu ke- 7 menyala
delay(500);
outportb (PortC,0x04); // lampu ke- 6 menyala
delay(500);
outportb (PortC,0x08); // lampu ke- 5 menyala
delay(500);
outportb (PortC,0x10); // lampu ke- 4 menyala
delay(500);
outportb (PortC,0x20); // lampu ke- 3 menyala
delay(500);
outportb (PortC,0x40); // lampu ke- 2 menyala
delay(500);
outportb (PortC,0x80); // lampu ke- 1 menyala
delay(500);
outportb (PortC,0xFF); // semua lampu menyala
delay(500);
outportb (PortC,0x00); // semua lampu padam
} while (!kbhit());
outportb (PortA,0x00);
outportb (PortB,0x00);
outportb (PortC,0x00);
return 0;
}

```

Setelah program dibuat maka langkah selanjutnya adalah menghubungkan ketiga *Port* PPI dengan Rangkaian I/O melalui kabel data. Selanjutnya, program dijalankan dan hasil pengujian dapat dilihat pada Tabel 4.1. *Output* yang diinginkan adalah lampu akan menyala secara berurutan dari lampu ke-8 sampai lampu ke-1 pada *Port* A hingga menyala semua lalu padam semua

dan berganti ke *Port B* dan *Port C* dengan urutan yang sama sampai lampu padam semua apabila ada penekanan tombol pada *keyboard*.

Tabel 4.1 Hasil pengujian modul PPI

Port	Nilai Heksa pada outport(port,heksa)	Lampu LED							
		1	2	3	4	5	6	7	8
A	FF H	1	1	1	1	1	1	1	0
B	FF H	1	1	1	1	1	1	1	1
C	FF H	1	1	1	1	1	1	1	1

Keterangan : 1 = Lampu LED nyala.  
0 = Lampu LED padam.

Analisa hasil pengujian pada ketiga *port* PPI yang ada, sebagai berikut, sesuai pada Tabel 4.1 diatas bahwa pada *Port A* pada lampu LED ke delapan padam yang seharusnya menyala, dari hasil tersebut maka *Port A* pada PPI yang digunakan pada tugas akhir ini tidaklah layak untuk digunakan, kemudian untuk *Port B* dan *Port C* layak digunakan karena *output*-nya sesuai.

#### 4.1.2 DAC

Pengujian pada DAC menggunakan Rangkaian I/O dan Multimeter, Rangkain I/O digunakan sebagai tegangan masukan dalam bentuk digital untuk DAC, sedangkan penunjukan besaran tegangan pada multimeter sebagai *output* dari DAC (sinyal analog). Sebelumnya dipastikan tegangan *input* Vcc sebesar +5V, tegangan Vref 12V, tegangan Vee -12V, dan juga *ground*. Karena input DAC ada 8 bit dan Vref yang digunakan adalah sebesar 12V, maka perubahan untuk setiap bit nya adalah  $\frac{12}{255} = 0.047V$ .

Selanjutnya pengujian di lanjutkan dengan menggeser – geser saklar *DIP Switch* pada rangkain I/O sebagai perwakilan dari perubahan bit kemudian di periksa *ouput* DAC dengan Multimeter yang menunjuk pada Voltmeter

Tabel 4.2 Hasil pengujian modul Rangkaian DAC

No	Saklar <i>DIP Switch</i> (Pin <i>input</i> DAC)								Tegangan <i>Output</i> (V)
	1 D7	2 D6	3 D5	4 D4	5 D3	6 D2	7 D1	8 D0	
1	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	1	0.047
3	0	0	0	0	0	0	1	1	0.14
4	0	0	0	0	0	1	1	1	0.33
5	0	0	0	0	1	1	1	1	0.7
6	0	0	0	1	1	1	1	1	1.46
7	0	0	1	1	1	1	1	1	3
8	0	1	1	1	1	1	1	1	5.9
9	1	1	1	1	1	1	1	1	11.98

Keterangan : 1 = Pada posisi *ON*  
0 = Pada posisi *OFF*

Analisa untuk hasil pengujian DAC yang ada pada Tabel 4.2 sebagai berikut, dengan  $V_{ref}$  12V maka perubahan tiap bitnya sebesar 0.047V, jadi bila tegangan *input* 0 desimal atau ke delapan saklar pada posisi *OFF* maka *ouput*-nya haruslah 0V, apabila kedelapan saklar pada posisi *ON* atau tegangan *input* 255

desimal, maka tegangan *output*-nya haruslah sebesar 12V. Dari Tabel 4.2 diatas hasil yang didapat sudah sesuai, maka DAC layak digunakan.

#### 4.1.3 ADC

Pengujian Rangkaian ADC dapat menggunakan sebuah *Potensio Multitune* dan Rangkaian I/O. Tapi sebelumnya, dipastikan terlebih dahulu nilai *clock*-nya sudah sebesar 600KHz dengan melihat besarnya Resistor dan Kapasitor yang terpasang pada *clock internal*. Selanjutnya menguji ADC dengan memutar *Potensio Multitune* yang terhubung ke tegangan 5V dan *ground* sebagai perwakilan dari perubahan nilai tegangan *input* ADC dan menggunakan Rangkaian I/O sebagai *output* dari ADC yang berbentuk tegangan digital.

Tabel 4.3 Hasil pengujian modul Rangkaian ADC

No	Tegangan Input (V)	Lampu LED (Pin <i>Output</i> ADC)							
		1 D7	2 D6	3 D5	4 D4	5 D3	6 D2	7 D1	8 D0
1.	0	0	0	0	0	0	0	0	0
2	0.020	0	0	0	0	0	0	0	1
3	0.039	0	0	0	0	0	0	1	0
4	0.079	0	0	0	0	0	1	0	0
5	0.160	0	0	0	0	1	0	0	0
6	0.319	0	0	0	1	0	0	0	0
7	0.641	0	0	1	0	0	0	0	0
8	1.279	0	1	0	0	0	0	0	0
9	2.550	1	0	0	0	0	0	0	0
10	5.100	1	1	1	1	1	1	1	1

Keterangan : 1 = Lampu Nyala.  
0 = Lampu Padam.

Analisa ke-3 pada rangkaian ADC, apabila *Potensio Multitune* diputar maksimal ke arah *ground* maka semua lampu LED di Rangkaian I/O haruslah padam semua, sebaliknya apabila *Potensio Multitune* diputar maksimal ke arah

5V lampu pada Rangkaian I/O harus menyala semua. Dari Tabel 4.3 dapat dilihat bahwa input dan output rangkaian ADC sudah sesuai, sehingga ADC layak untuk digunakan.

#### 4.1.4 Driver Motor DC

Pengujian *Driver* Motor DC ditekankan pada pengukuran arus sebelum dan sesudah memasuki rangkaian *Driver*, karena mengingat fungsinya disini hanya sebatas untuk menguatkan arus dari tegangan yang dihasilkan oleh DAC untuk tegangan *input* Motor DC. Cara pengujian *Driver* Motor DC dapat menggunakan Multimeter dengan menunjuk pada Amperemeter, titik yang diuji adalah *ouput* dari DAC serta *ouput* dari *Driver* motor DC.

Tabel 4.4 Hasil pengukuran arus pada *Driver* Motor DC

Tegangan <i>Input</i> (V)	Motor DC 1		Motor DC 3	
	Arus <i>Output</i> DAC (mA)	Arus <i>Output</i> <i>Driver</i> (A)	Arus <i>Output</i> DAC (mA)	Arus <i>Output</i> <i>Driver</i> (A)
1	0	0	0	0
2	0	0	0	0
3	0	0	0	0
4	1.53	0.15	2.0	0.18
5	1.53	0.15	2.0	0.18
6	1.56	0.15	2.1	0.19
7	1.61	0.15	2.2	0.20
8	1.68	0.16	2.3	0.20
9	1.73	0.16	2.4	0.21
10	1.79	0.16	2.5	0.22
11	1.84	0.17	2.6	0.23
12	1.85	0.17	2.7	0.23

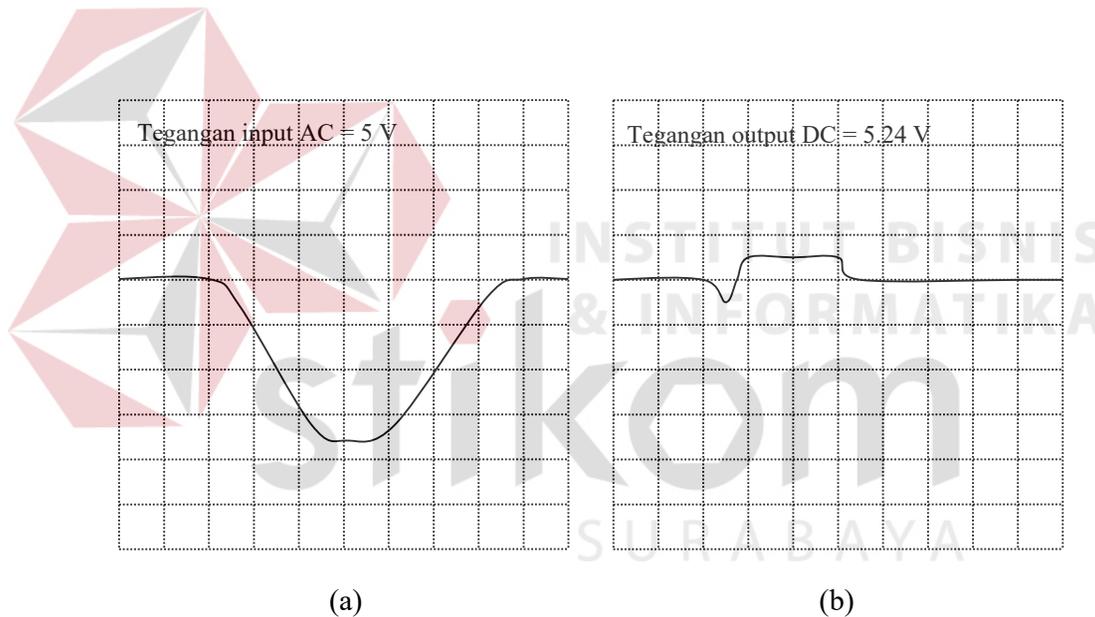
Keterangan : 0 (nol), karena Motor DC tidak berputar

Analisa untuk *Driver* Motor DC ditekankan pada penguatan arus sebelum dan sesudah melewati rangkaian *Driver* Motor DC. Dari Tabel 4.4 telah

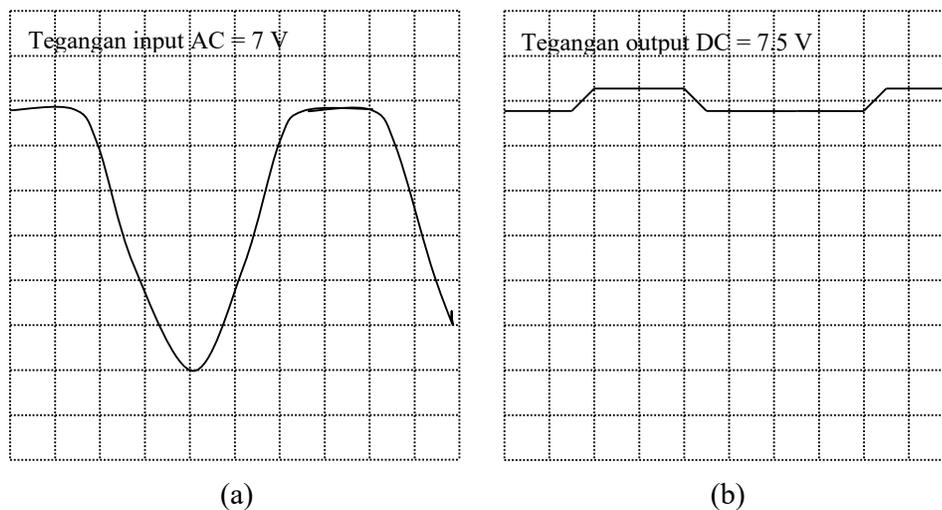
di dapatkan penguatan arus sebesar  $\pm 100$  kali dari tegangan *output* DAC (*input* Motor DC). Sehingga Rangkaian *Driver* sudah layak untuk digunakan.

#### 4.1.5 Rangkaian penyearah

Pengujian pada Rangkaian Penyearah dilakukan dengan menggunakan Osiloskop yang ada di Laboratorium Kontrol serta Multimeter. Penggunaan Osiloskop untuk melihat bentuk sinyal tegangan sebelum dan sesudah masuk ke Rangkaian Penyearah, sedangkan Multimeter untuk melihat jenis dan besarnya tegangan sebelum dan sesudah melewati Rangkaian Penyearah.



Gambar 4.1 Bentuk gelombang *output Tachogenerator* dari Motor DC 1, (a) sebelum melalui Rangkaian Penyearah (b) sesudah melalui Rangkaian Penyearah



Gambar 4.2 Bentuk gelombang *output Tachogenerator* dari Motor DC 3, (a) sebelum melalui Rangkaian Penyearah (b) sesudah melalui Rangkaian Penyearah

Dari hasil pengujian pada Gambar 4.1 dan 4.2 dapat dilihat bahwa gelombang sinusoida tegangan AC sebelumnya masuk ke Rangkaian Penyearah telah berubah menjadi garis lurus dengan riak kecil atau menjadi tegangan DC yang sudah siap dipakai untuk proses selanjutnya. Begitu juga dengan tegangan yang terukur pada Multimeter, telah mengalami perubahan dan tegangan menjadi lebih besar, hal tersebut tidak mempengaruhi hasil akhir dari alat karena tegangan *output* rangkaian akan turun lagi bila diberi beban kerja. Sehingga dapat disimpulkan bahwa Rangkaian Penyearah layak digunakan.

#### 4.1.6 Rangkaian pelemah

Pengujian untuk Rangkaian Pelemah dapat menggunakan Multimeter, pengujian dilakukan seperti berikut, pertama pada titik *input* diberi tegangan masukan dan dilihat di Multimeter apakah sudah sesuai, kemudian di titik kedua pada *output* pelemah pertama yang sekaligus juga merupakan *input* untuk

pelemah kedua, dilihat apakah sudah didapat nilai yang lebih kecil dan negatif, pada titik ke tiga yaitu *output* dari pelemah ke dua dilihat apakah sudah didapat nilai yang sama pada pengukuran di titik ke dua dan bernilai positif.

Tabel 4.5 Hasil pengujian modul Rangkaian Pelemah

No	Tegangan <i>Input</i> (V)	Tegangan <i>Output</i> (V)	Pelemahan
1	7.01	2.95	2.38
2	5.98	2.5	2.39
3	4.99	2.08	2.4
4	4.02	1.67	2.41
5	2.97	1.25	2.37
Total			11.95
Total / Jumlah Data			2.39

Dari hasil pengujian pada Rangkaian Pelemah di Tabel 4.5 didapatkan bahwa nilai *ouput* dari rangkaian sudah mengalami pelemahan sebesar  $\pm 2.4$  kali dari tegangan yang dimasukkan ke rangkaian. Dari total kolom pelemahan dibagi dengan jumlah pengujian didapatkan nilai 2.39, sehingga dapat disimpulkan bahwa Rangkaian Pelemah layak digunakan.

#### 4.1.7 *Tachogenerator*

Pengujian pada *Tachogenerator* dapat menggunakan Multimeter dan RPMmeter pada Laboratorium Kontrol. Multimeter digunakan untuk mengukur tegangan yang keluar dari *Tachogenerator*, sedangkan RPMmeter untuk mengukur jumlah putaran pada *input Tachogenerator*. *Tachogenerator* dihubungkan dengan sebuah Motor DC yang diberi tegangan tertentu sehingga output dari *Tachogenerator* berupa tegangan dapat terbaca, kemudian diukur putaran *Tachogenerator* dengan RPMmeter

Tabel 4.6 Hasil pengujian *Tachogenerator*

No	Putaran (rpm)	Output (Volt)
1	850	2.4
2	1500	3.6
3	1750	4.9
4	2250	6.1
5	2700	6.9
6	3100	7.6

Analisa pada *Tachogenerator* didasarkan pada gambar karakteristik *Tachogenerator* dari Servo Fundamental Trainer yang menjelaskan bahwa untuk masukan sebesar  $\pm 1000$  rpm pada *Tachogenerator* akan menghasilkan tegangan sebesar 2,5 Volt. Berdasarkan Tabel 4.6 maka *Tachogenerator* yang dipakai sudah layak digunakan sebagai sensor kecepatan putaran pada Motor DC.

## 4.2 Prosedur Pengujian Perangkat Lunak

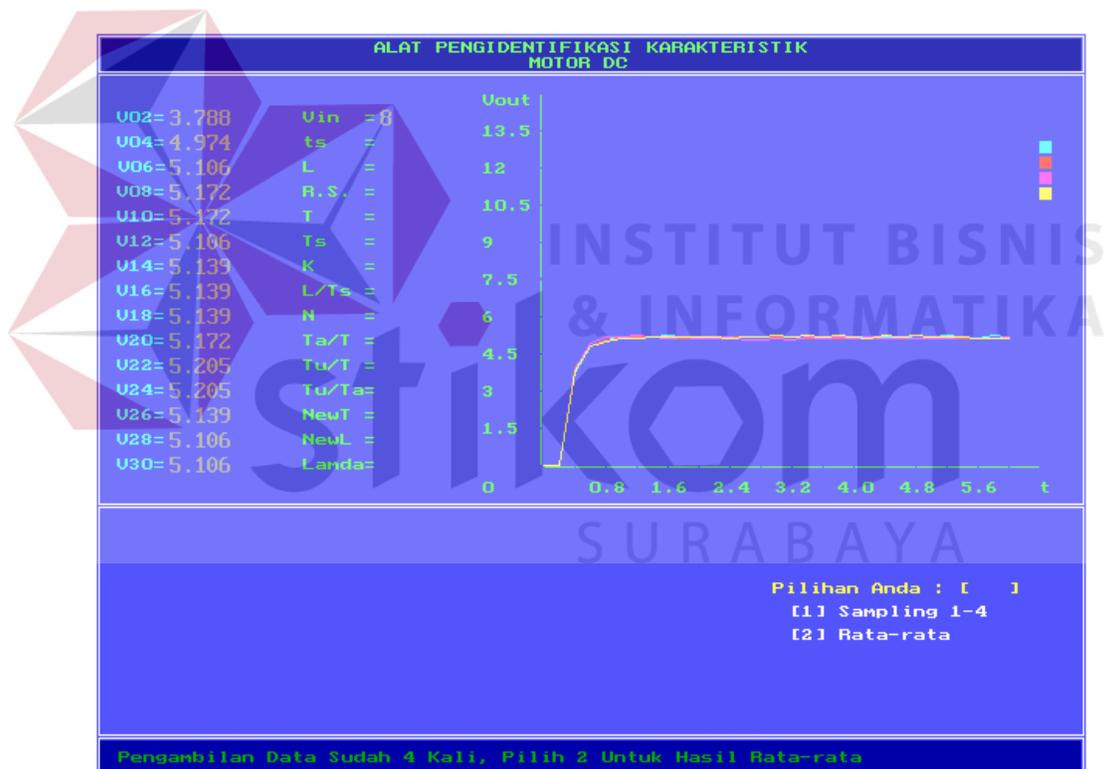
Pengujian perangkat lunak dilakukan dengan melakukan pengujian yang berkaitan dengan setiap bagian prosedur program, seperti pada bagian ambil data, rata – rata data, menampilkan data, menganalisa data dan lain sebagainya. Untuk tiap bagian dapat kita uji secara bersamaan dengan program utama ataupun secara terpisah.

### 4.2.1 Identifikasi secara statis

Untuk program identifikasi secara statis ada 3 prosedur penting yang dibuat. Ketiga prosedur tersebut adalah mengambil data, merata – rata data kemudian menganalisa data sampai mendapatkan fungsi alih yang dimaksud.

## A. Ambil data

Pengujian pada prosedur ambil data dilakukan dengan mengambil data dari Port C PPI sebanyak yang diperlukan, dengan terlebih dahulu memberikan *input*-an berupa *unit step* ke Motor, dan dilanjutkan dengan menyimpannya pada sebuah file dengan format yang sudah ditentukan. Penyimpanan data dilakukan pada 2 file yang terpisah. File pertama digunakan sebagai media penyimpanan *database*, sedangkan file ke dua digunakan untuk file *output* program yang nantinya dapat diakses dengan Microsoft Excel.



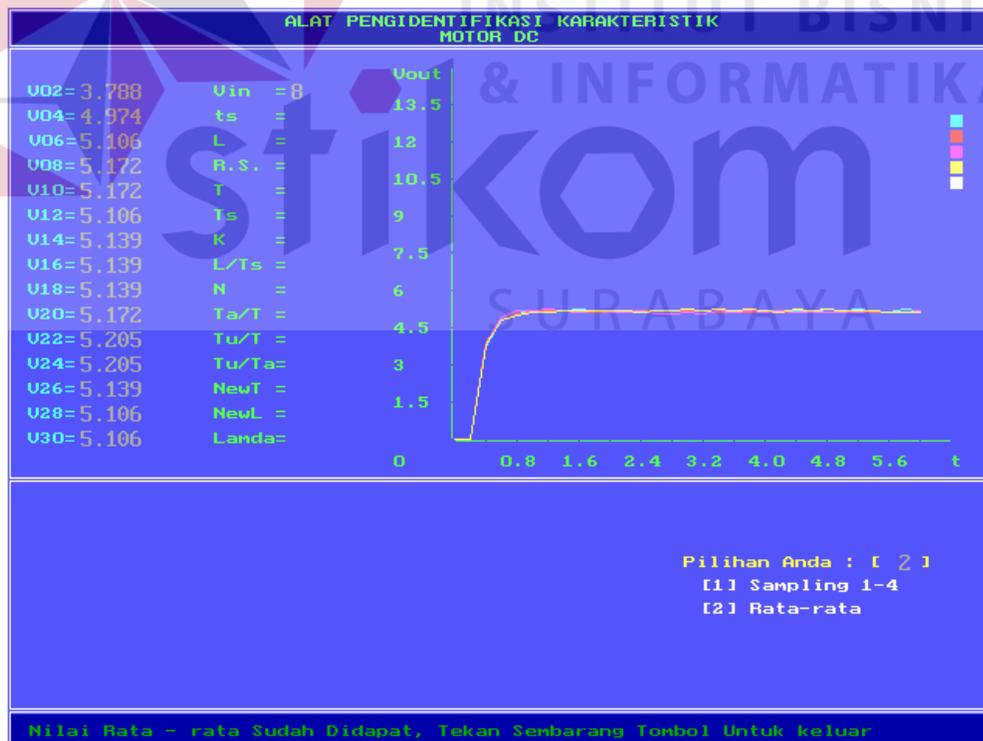
Gambar 4.3 Hasil prosedur ambil data untuk identifikasi secara statis.

Pada prosedur ambil data dengan periode *sampling* sebesar 0.2 detik, masih diketemukannya *error* periode *sampling* berkisar 0.03 detik, yang dapat menyebabkan program *crash*, walaupun hal tersebut sangat jarang terjadi. Tetapi

program yang dibuat sudah dapat mengambil data dan menampilkannya mendekati *real time*. Sehingga prosedur program yang dibuat sudah sesuai dan layak digunakan.

## B. Rata – rata data

Dalam pengujian terhadap prosedur rata – rata, program dibuat untuk dapat membaca file *database*, mencari nilai – nilai tiap percobaan dan menghitung rata – rata tiap data *sampling*, kemudian menambahkan hasil rata – rata ke dalam file *database* yang sama sebagai data ke-5. File ini nantinya akan diakses untuk dapat dianalisa dan ditampilkan nilai – nilainya hingga dapat membangun sebuah fungsi alih seperti yang diinginkan.

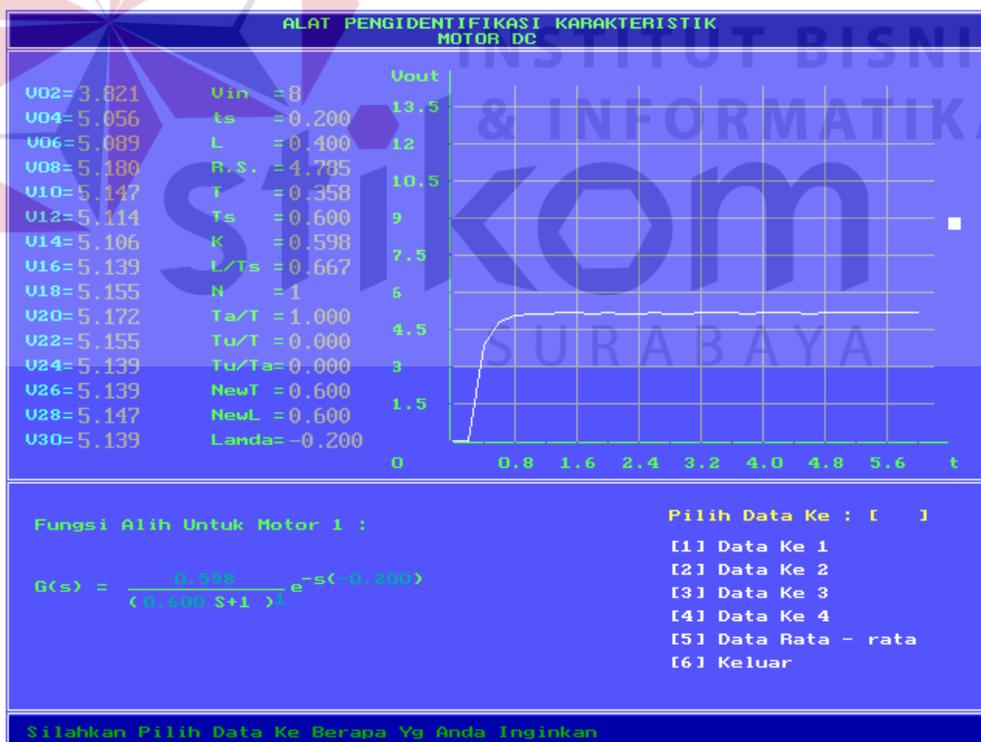


Gambar 4.4 Hasil prosedur rata - rata data untuk identifikasi secara statis.

Untuk prosedur rata – rata data, program yang dibuat dapat membaca file *database* dan dapat mencari rata – rata data untuk dijadikan data ke-5. sehingga program yang dibuat sudah sesuai dan layak digunakan.

### C. Analisa data

Pada prosedur ini banyak sekali proses yang akan dilakukan, sehingga pengujian dilakukan secara bertahap sesuai urutan dari identifikasi statis. mulai dari membaca data percobaan beberapa yang diinginkan, kemudian data percobaan yang diinginkan dari data ke - 1 sampai data ke - 30 diambil. Kemudian dicari nilai – nilai yang membangun suatu fungsi alih, sesuai dari urutan identifikasi.

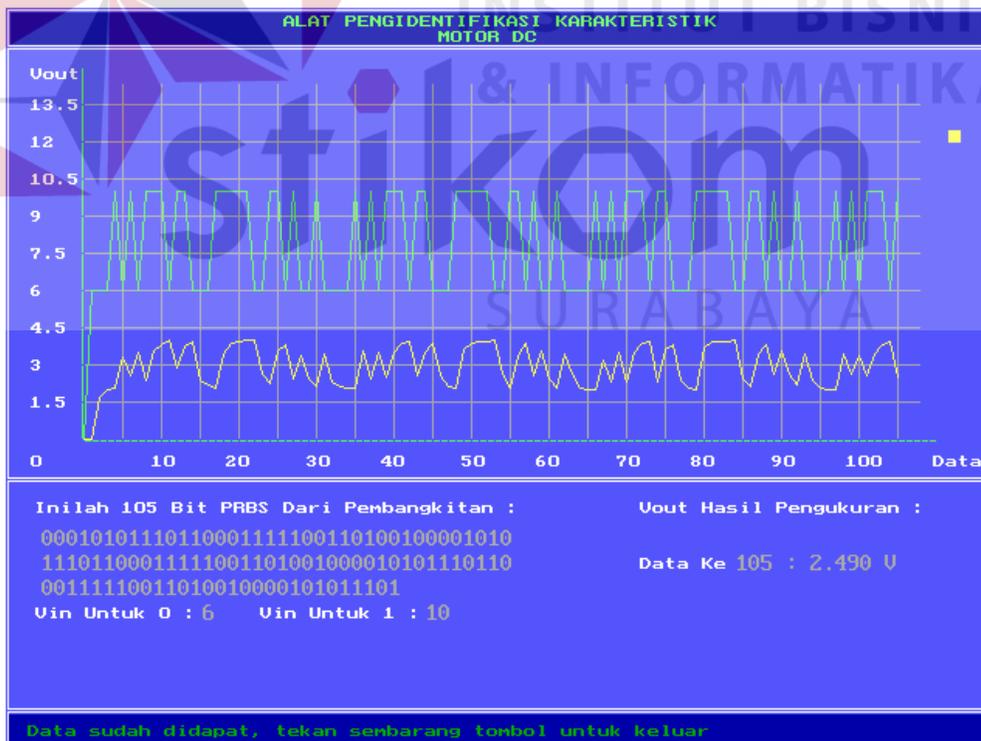


Gambar 4.5 Hasil prosedur analisa beserta hasilnya untuk identifikasi statis.

Prosedure terakhir dari identifikasi secara statis adalah menganalisa data yang diperoleh baik data ke-1 sampai data ke-4 ataupun data hasil rata – rata menurut algoritma pada identifikasi secara statis. Dari gambar 4.5 diatas dapat dilihat bahwa nilai – nilai untuk membangun sebuah fungsi alih sudah didapat dan juga fungsi alihnya sudah ditampilkan. Sehingga program untuk prosedur ini sudah layak untuk digunakan.

#### 4.2.2 Identifikasi secara dinamis

Pengujian pada program identifikasi secara dinamis hanya sebatas pada pengujian prosedur mengambil data sesuai *input*-an PRBS yang diberikan. Karena program yang dibuat dengan hanya untuk mengambil data, sedangkan untuk analisa *output*-nya menggunakan program MATLAB.



Gambar 4.6 Hasil prosedur mengambil data untuk identifikasi secara dinamis.

Untuk program identifikasi secara dinamis sudah layak digunakan, karena dapat dilihat pada Gambar 4.6 diatas *output* dari Motor DC sudah sesuai dengan *input*-an tegangan yang berupa PRBS.

### 4.3 Pengujian alat yang terintegrasi

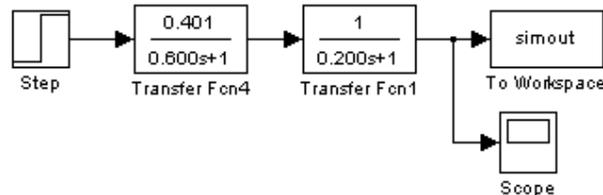
Pengujian disini berarti pengujian penuh pada alat yang dibuat dengan mengintegrasikan perangkat keras dengan perangkat lunak yang telah dibuat. Pengujian dilakukan dengan menjalankan program dan dilihat apakah *output* program ataupun hasil pada perangkat keras sudah sesuai dengan yang dikehendaki. Apabila sudah sesuai, maka pengujian ini dilakukan secara berulang - ulang dengan tegangan *input* yang berbeda – beda.

Dalam pengujian ini akan diambil 10 data pada tiap tegangan input, 8V, 9V, 10V, 11V dan 12V dengan *time sampling* sebesar 0.2 detik dengan menggunakan identifikasi secara statis.

Untuk mengetahui keabsahan fungsi alih dari percobaan maka dilakukan analisa dengan bantuan SIMULINK pada program MATLAB 6.5. Dengan memberikan *input* berupa *unit step* pada fungsi alih yang di simulasikan akan di dapat tegangan *output*-nya, kemudian akan dibandingkan dengan tegangan *output* yang dari Motor DC. Dari selisih tiap nilai yang dibandingkan akan diketemukan *Norm Error*, yang nilainya dapat mewakili *error* dari fungsi alih yang di dapat.

Sebagai contoh, akan disimulasi fungsi alih  $G(s) = \frac{0.401}{(0.600s + 1)^1} e^{-s(-0.200)}$ , dengan tegangan input 9V, periode *sampling* 0.2 detik dan hasilnya akan ditampung di variabel simout. Untuk *unit step*, *option* diisikan sebagai berikut, *step time* = 0.2, *initial value* = 0, *final value* = 9, *sample time* = -

1. Untuk *simout option*-nya sebagai berikut, *decimation* = 1, *sample time* = 0.2, dan *output*-nya berbentuk *array*. Untuk nilai  $e^{-s(-0.200)}$  dapat dirubah menjadi  $\frac{1}{0.2s+1}$  sesuai dengan *Transport lag*.



Gambar 4.7 Simulasi suatu fungsi alih dalam *Simulink*

Algoritma program untuk mencari norm error dalam MATLAB sebagai berikut :

```
Out1 = FILE_OUTPUT_MOTOR(1:30,kolom);
Out2 = FILE_OUTPUT_SIMULASI(1:30,kolom);
Norm(out1 - out2);
```

*Out1* adalah nilai – nilai di kolom pada nama file *output* Motor DC, yang didapat dari hasil program dalam C++, kemudian *Out2* adalah nilai – nilai di kolom pada nama file *output* hasil simulasi, yang didapat dari variabel *simout* hasil dari SIMULINK pada MATLAB. Hasil simulasi bisa dilihat dalam grafik ini pada Gambar 4.7 dan Tabel 4.13 dibawah.

Tabel 4.7 Hasil pengujian alat dengan *input* 8V, *ts* = 0.2 detik

No.	Motor 1	Motor 2	Motor 3
1	$\frac{0.420}{(0.314s+1)^4} e^{-s(-0.582)}$	$\frac{0.386}{(0.800s+1)^1} e^{-s(-0.400)}$	$\frac{0.583}{(0.600s+1)^1} e^{-s(-0.200)}$

2	$\frac{0.417}{(0.314s + 1)^4} e^{-s(-0.582)}$	$\frac{0.398}{(0.600s + 1)^1} e^{-s(-0.200)}$	$\frac{0.562}{(0.600s + 1)^1} e^{-s(-0.200)}$
3	$\frac{0.419}{(0.800s + 1)^1} e^{-s(-0.400)}$	$\frac{0.394}{(0.134s + 1)^4} e^{-s(-0.221)}$	$\frac{0.571}{(0.600s + 1)^1} e^{-s(-0.200)}$
4	$\frac{0.418}{(0.314s + 1)^4} e^{-s(-0.582)}$	$\frac{0.394}{(0.600s + 1)^1} e^{-s(-0.200)}$	$\frac{0.573}{(0.600s + 1)^1} e^{-s(-0.200)}$
5	$\frac{0.416}{(0.269s + 1)^4} e^{-s(-0.442)}$	$\frac{0.385}{(0.314s + 1)^4} e^{-s(-0.582)}$	$\frac{0.578}{(0.600s + 1)^1} e^{-s(-0.200)}$
6	$\frac{0.419}{(0.600s + 1)^1} e^{-s(-0.200)}$	$\frac{0.392}{(0.600s + 1)^1} e^{-s(-0.200)}$	$\frac{0.578}{(0.600s + 1)^1} e^{-s(-0.200)}$
7	$\frac{0.420}{(0.600s + 1)^1} e^{-s(-0.200)}$	$\frac{0.386}{(0.600s + 1)^1} e^{-s(-0.200)}$	$\frac{0.578}{(0.600s + 1)^1} e^{-s(-0.200)}$
8	$\frac{0.418}{(0.314s + 1)^4} e^{-s(-0.582)}$	$\frac{0.388}{(0.800s + 1)^1} e^{-s(-0.400)}$	$\frac{0.567}{(0.600s + 1)^1} e^{-s(-0.200)}$
9	$\frac{0.416}{(0.487s + 1)^3} e^{-s(-1.836)}$	$\frac{0.388}{(0.600s + 1)^1} e^{-s(-0.200)}$	$\frac{0.571}{(0.600s + 1)^1} e^{-s(-0.200)}$
10	$\frac{0.417}{(0.487s + 1)^3} e^{-s(-1.836)}$	$\frac{0.391}{(0.800s + 1)^1} e^{-s(-0.400)}$	$\frac{0.580}{(0.600s + 1)^1} e^{-s(-0.200)}$

Tabel 4.8 Hasil pengujian alat dengan *input* 9V,  $t_s = 0.2$  detik

No.	Motor 1	Motor 2	Motor 3
1	$\frac{0.432}{(0.269s + 1)^4} e^{-s(-0.442)}$	$\frac{0.413}{(0.600s + 1)^1} e^{-s(-0.200)}$	$\frac{0.592}{(0.600s + 1)^1} e^{-s(-0.200)}$
2	$\frac{0.430}{(0.314s + 1)^4} e^{-s(-0.582)}$	$\frac{0.408}{(0.600s + 1)^1} e^{-s(-0.200)}$	$\frac{0.591}{(0.600s + 1)^1} e^{-s(-0.200)}$
3	$\frac{0.429}{(1.030s + 1)^1} e^{-s(-9.529)}$	$\frac{0.411}{(0.541s + 1)^3} e^{-s(-2.084)}$	$\frac{0.586}{(0.134s + 1)^4} e^{-s(-0.221)}$
4	$\frac{0.430}{(0.269s + 1)^4} e^{-s(-0.442)}$	$\frac{0.410}{(0.800s + 1)^1} e^{-s(-0.400)}$	$\frac{0.590}{(0.600s + 1)^1} e^{-s(-0.200)}$
5	$\frac{0.431}{(0.600s + 1)^1} e^{-s(-0.200)}$	$\frac{0.409}{(0.800s + 1)^1} e^{-s(-0.400)}$	$\frac{0.595}{(0.600s + 1)^1} e^{-s(-0.200)}$
6	$\frac{0.430}{(0.195s + 1)^5} e^{-s(-0.076)}$	$\frac{0.411}{(0.314s + 1)^4} e^{-s(-0.582)}$	$\frac{0.591}{(0.600s + 1)^1} e^{-s(-0.200)}$
7	$\frac{0.430}{(0.487s + 1)^3} e^{-s(-1.836)}$	$\frac{0.410}{(0.600s + 1)^1} e^{-s(-0.200)}$	$\frac{0.592}{(0.134s + 1)^4} e^{-s(-0.221)}$
8	$\frac{0.433}{(0.269s + 1)^4} e^{-s(-0.442)}$	$\frac{0.415}{(0.600s + 1)^1} e^{-s(-0.200)}$	$\frac{0.601}{(0.134s + 1)^4} e^{-s(-0.221)}$
9	$\frac{0.431}{(0.600s + 1)^1} e^{-s(-0.200)}$	$\frac{0.414}{(0.600s + 1)^1} e^{-s(-0.200)}$	$\frac{0.608}{(0.134s + 1)^4} e^{-s(-0.221)}$

10	$\frac{0.433}{(0.487s + 1)^3} e^{-s(-1.836)}$	$\frac{0.409}{(0.600s + 1)^1} e^{-s(-0.200)}$	$\frac{0.599}{(0.600s + 1)^1} e^{-s(-0.200)}$
----	---	---	---

Tabel 4.9 Hasil pengujian alat dengan *input* 10V,  $t_s = 0.2$  detik

No.	Motor 1	Motor 2	Motor 3
1	$\frac{0.442}{(0.269s + 1)^4} e^{-s(-0.442)}$	$\frac{0.421}{(0.600s + 1)^1} e^{-s(-0.200)}$	$\frac{0.618}{(0.600s + 1)^1} e^{-s(-0.200)}$
2	$\frac{0.443}{(0.800s + 1)^1} e^{-s(-0.400)}$	$\frac{0.422}{(0.600s + 1)^1} e^{-s(-0.200)}$	$\frac{0.610}{(0.134s + 1)^4} e^{-s(-0.221)}$
3	$\frac{0.441}{(0.600s + 1)^1} e^{-s(-0.200)}$	$\frac{0.426}{(0.600s + 1)^1} e^{-s(-0.200)}$	$\frac{0.613}{(0.134s + 1)^4} e^{-s(-0.221)}$
4	$\frac{0.445}{(0.600s + 1)^1} e^{-s(-0.200)}$	$\frac{0.426}{(0.541s + 1)^3} e^{-s(-2.084)}$	$\frac{0.607}{(0.600s + 1)^1} e^{-s(-0.200)}$
5	$\frac{0.442}{(0.487s + 1)^3} e^{-s(-1.836)}$	$\frac{0.428}{(0.433s + 1)^3} e^{-s(-1.587)}$	$\frac{0.607}{(0.600s + 1)^1} e^{-s(-0.200)}$
6	$\frac{0.446}{(0.600s + 1)^1} e^{-s(-0.200)}$	$\frac{0.428}{(0.195s + 1)^5} e^{-s(-0.076)}$	$\frac{0.601}{(0.600s + 1)^1} e^{-s(-0.200)}$
7	$\frac{0.444}{(0.595s + 1)^3} e^{-s(-2.332)}$	$\frac{0.431}{(0.600s + 1)^1} e^{-s(-0.200)}$	$\frac{0.672}{(0.662s + 1)^2} e^{-s(-0.183)}$
8	$\frac{0.445}{(0.800s + 1)^1} e^{-s(-0.400)}$	$\frac{0.432}{(0.433s + 1)^3} e^{-s(-1.587)}$	$\frac{0.617}{(0.600s + 1)^1} e^{-s(-0.200)}$

9	$\frac{0.445}{(0.800s + 1)^1} e^{-s(-0.400)}$	$\frac{0.432}{(0.314s + 1)^4} e^{-s(-0.582)}$	$\frac{0.671}{(0.314s + 1)^4} e^{-s(-0.582)}$
10	$\frac{0.445}{(0.800s + 1)^1} e^{-s(-0.400)}$	$\frac{0.433}{(0.800s + 1)^1} e^{-s(-0.400)}$	$\frac{0.610}{(0.134s + 1)^4} e^{-s(-0.221)}$

Tabel 4.10. Hasil pengujian alat dengan *input* 11V,  $t_s = 0.2$  detik

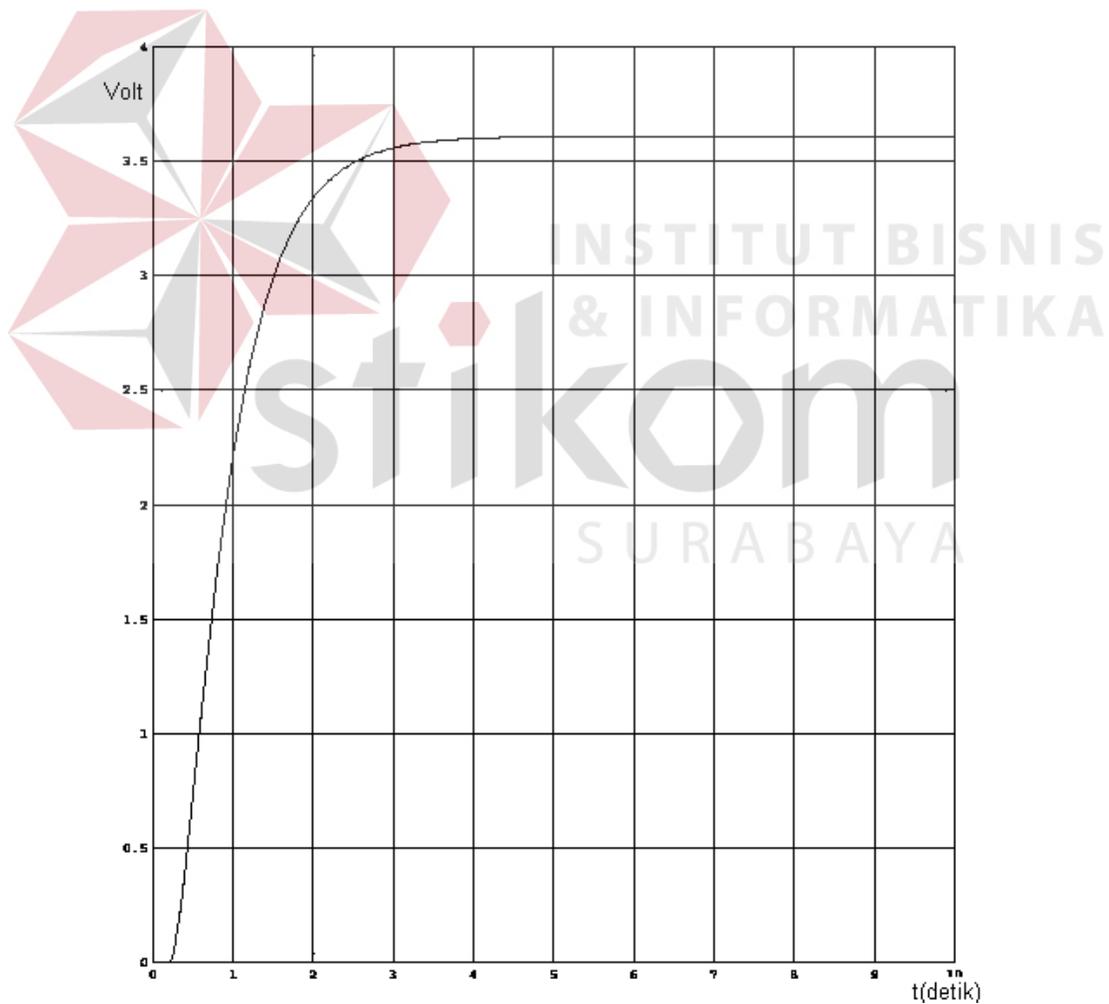
No.	Motor 1	Motor 2	Motor 3
1	$\frac{0.457}{(0.600s + 1)^1} e^{-s(-0.200)}$	$\frac{0.447}{(0.433s + 1)^3} e^{-s(-1.587)}$	$\frac{0.618}{(0.134s + 1)^4} e^{-s(-0.221)}$
2	$\frac{0.455}{(0.600s + 1)^1} e^{-s(-0.200)}$	$\frac{0.446}{(0.433s + 1)^3} e^{-s(-1.587)}$	$\frac{0.614}{(0.134s + 1)^4} e^{-s(-0.221)}$
3	$\frac{0.454}{(0.600s + 1)^1} e^{-s(-0.200)}$	$\frac{0.450}{(0.269s + 1)^3} e^{-s(-0.442)}$	$\frac{0.608}{(0.600s + 1)^1} e^{-s(-0.200)}$
4	$\frac{0.454}{(0.595s + 1)^3} e^{-s(-0.200)}$	$\frac{0.448}{(0.600s + 1)^1} e^{-s(-0.200)}$	$\frac{0.603}{(0.600s + 1)^1} e^{-s(-0.200)}$
5	$\frac{0.451}{(0.134s + 1)^4} e^{-s(-0.221)}$	$\frac{0.448}{(0.600s + 1)^1} e^{-s(-0.200)}$	$\frac{0.603}{(0.134s + 1)^4} e^{-s(-0.221)}$
6	$\frac{0.452}{(0.314s + 1)^4} e^{-s(-0.582)}$	$\frac{0.445}{(0.600s + 1)^1} e^{-s(-0.200)}$	$\frac{0.610}{(0.134s + 1)^4} e^{-s(-0.221)}$
7	$\frac{0.453}{(0.600s + 1)^1} e^{-s(-0.200)}$	$\frac{0.448}{(0.800s + 1)^1} e^{-s(-0.400)}$	$\frac{0.606}{(0.600s + 1)^1} e^{-s(-0.200)}$
8	$\frac{0.453}{(0.600s + 1)^1} e^{-s(-0.200)}$	$\frac{0.448}{(0.800s + 1)^1} e^{-s(-0.400)}$	$\frac{0.603}{(0.134s + 1)^4} e^{-s(-0.221)}$

9	$\frac{0.455}{(0.800s + 1)^1} e^{-s(-0.400)}$	$\frac{0.445}{(0.600s + 1)^1} e^{-s(-0.200)}$	$\frac{0.595}{(0.600s + 1)^1} e^{-s(-0.200)}$
10	$\frac{0.452}{(0.800s + 1)^1} e^{-s(-0.400)}$	$\frac{0.445}{(0.800s + 1)^1} e^{-s(-0.400)}$	$\frac{0.597}{(0.600s + 1)^1} e^{-s(-0.200)}$

Tabel 4.11 Hasil pengujian alat dengan *input* 12V,  $t_s = 0.2$  detik

No.	Motor 1	Motor 2	Motor 3
1	$\frac{0.423}{(0.800s + 1)^1} e^{-s(-0.400)}$	$\frac{0.414}{(0.134s + 1)^4} e^{-s(-0.221)}$	$\frac{0.559}{(0.600s + 1)^1} e^{-s(-0.200)}$
2	$\frac{0.426}{(0.600s + 1)^1} e^{-s(-0.200)}$	$\frac{0.417}{(0.195s + 1)^5} e^{-s(-0.076)}$	$\frac{0.556}{(0.600s + 1)^1} e^{-s(-0.200)}$
3	$\frac{0.421}{(0.134s + 1)^4} e^{-s(-0.221)}$	$\frac{0.417}{(0.600s + 1)^1} e^{-s(-0.200)}$	$\frac{0.561}{(0.134s + 1)^4} e^{-s(-0.221)}$
4	$\frac{0.423}{(0.600s + 1)^1} e^{-s(-0.200)}$	$\frac{0.417}{(0.600s + 1)^1} e^{-s(-0.200)}$	$\frac{0.562}{(0.600s + 1)^1} e^{-s(-0.200)}$
5	$\frac{0.427}{(0.600s + 1)^1} e^{-s(-0.200)}$	$\frac{0.418}{(0.600s + 1)^1} e^{-s(-0.200)}$	$\frac{0.561}{(0.134s + 1)^4} e^{-s(-0.221)}$
6	$\frac{0.425}{(0.314s + 1)^4} e^{-s(-0.582)}$	$\frac{0.418}{(0.314s + 1)^4} e^{-s(-0.582)}$	$\frac{0.564}{(0.134s + 1)^4} e^{-s(-0.221)}$
7	$\frac{0.425}{(0.800s + 1)^1} e^{-s(-0.400)}$	$\frac{0.418}{(0.541s + 1)^3} e^{-s(-2.084)}$	$\frac{0.558}{(0.134s + 1)^4} e^{-s(-0.221)}$

8	$\frac{0.426}{(0.800s + 1)^1} e^{-s(-0.400)}$	$\frac{0.418}{(0.800s + 1)^1} e^{-s(-0.400)}$	$\frac{0.559}{(0.134s + 1)^4} e^{-s(-0.221)}$
9	$\frac{0.423}{(0.314s + 1)^4} e^{-s(-0.582)}$	$\frac{0.418}{(0.134s + 1)^4} e^{-s(-0.221)}$	$\frac{0.557}{(0.134s + 1)^4} e^{-s(-0.221)}$
10	$\frac{0.429}{(0.800s + 1)^1} e^{-s(-0.400)}$	$\frac{0.417}{(0.195s + 1)^5} e^{-s(-0.076)}$	$\frac{0.560}{(0.134s + 1)^4} e^{-s(-0.221)}$



Gambar 4.8 Grafik dari Scope pada Simulink.



Tabel 4.13 Hasil *output* dari Motor DC 1 dan simulasi pada Motor DC 1 pada tegangan *input* 9V, untuk menemukan *norm error*.

No	Output		Selisih <sup>2</sup>	No	Output		Selisih <sup>2</sup>
	Motor DC	Simulasi			Motor DC	Simulasi	
1	0	0	0	16	3.673	3.5099	0.0266
2	0.348	0	0.1211	17	3.656	3.5380	0.0139
3	3.608	0.3939	10.3304	18	3.648	3.5829	0.0042
4	3.689	1.0739	6.8387	19	3.656	3.5903	0.0043
5	3.697	1.7074	4.6884	20	3.665	3.5956	0.0048
6	3.664	2.215	2.0996	21	3.673	3.5994	0.0054
7	3.681	2.5987	1.1714	22	3.624	3.6021	0.0005
8	3.648	2.8809	0.5884	23	3.664	3.6041	0.0036
9	3.640	3.0857	0.3072	24	3.689	3.6055	0.0070
10	3.648	3.2335	0.1718	25	3.664	3.6065	0.0033
11	3.640	3.3397	0.0902	26	3.648	3.6072	0.0017
12	3.665	3.4160	0.0620	27	3.648	3.6077	0.0016
13	3.648	3.4707	0.0314	28	3.665	3.6081	0.0032
14	3.705	3.5099	0.0381	29	3.615	3.6083	0.0001
15	3.656	3.5380	0.0139	30	3.665	3.6085	0.0032
Total1			26.5526	Total2			0.0792

Dari Tabel 4.13 dapat di cari *norm error*-nya dengan cara, Total = Total1 + Total2 = 26.5526+0.0792 = 26.6318. Kemudian dari Total yang didapat

dicari akar kuadratnya,  $\sqrt{26.6318} = 5.1606$ . Dari *norm error* yang didapat, dapat disimpulkan bahwa fungsi alihnya layak untuk digunakan.

Jika nilai *norm error*  $\leq 10$  maka fungsi alih yang di dapat layak untuk digunakan. Makin besar *norm error* makin jauh perbedaan antara output hasil simulasi dengan hasil identifikasi.

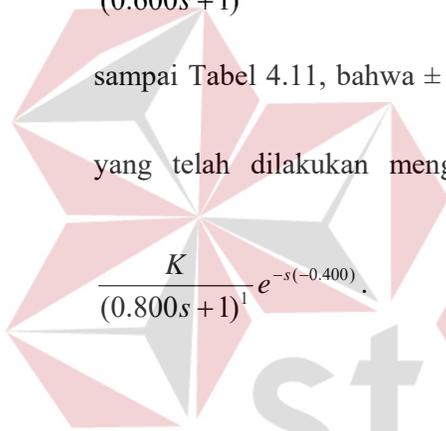
Ditinjau dari besarnya *norm error*, dari Tabel 4.12 dapat disimpulkan bahwa Fungsi alih yang mewakili karakteristik Motor DC adalah

$\frac{K}{(0.600s + 1)^1} e^{-s(-0.200)}$  dan  $\frac{K}{(0.800s + 1)^1} e^{-s(-0.400)}$ . Sedangkan dari Tabel 4.7

sampai Tabel 4.11, bahwa  $\pm 50\%$  dari hasil keseluruhan pelaksanaan identifikasi

yang telah dilakukan mengacu pada fungsi alih  $\frac{K}{(0.600s + 1)^1} e^{-s(-0.200)}$  dan

$\frac{K}{(0.800s + 1)^1} e^{-s(-0.400)}$ .



INSTITUT BISNIS  
& INFORMATIKA

stikom

SURABAYA